小白成长日记——第十七堆叠表单

以下实例使用两个输入框,一个复选框,一个提交按钮来创建堆叠表单:天 我变强了,也要秃了

bootstrap

1. 表单

堆叠表单: 以下实例使用两个输入框,一个复选框,一个提交按钮来创建堆叠表单;

```
<!DOCTYPE html>
<html>
<head>
<title>Bootstrap4 实例</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"</pre>
href="https://cdn.bootcss.com/bootstrap/4.1.0/css/bootstrap.min.css">
<script src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js">
</script>
<script
src="https://cdn.bootcss.com/popper.js/1.12.5/umd/popper.min.js">
</script>
<script
src="https://cdn.bootcss.com/bootstrap/4.1.0/js/bootstrap.min.js">
</script>
</head>
<body>
<div class="container">
<h2>堆叠表单</h2>
<form>
 <div class="form-group">
  <label for="email">Email:</label>
  <input type="email" class="form-control" id="email"</pre>
placeholder="Enter email">
 </div>
 <div class="form-group">
  <label for="pwd">Password:</label>
  <input type="password" class="form-control" id="pwd"</pre>
placeholder="Enter password">
 </div>
 <div class="form-check">
  <label class="form-check-label">
   <input class="form-check-input" type="checkbox"> Remember me
  </label>
  </div>
 <button type="submit" class="btn btn-primary">Submit</button>
```

```
</form>
</div>
</body>
</html>
```

内联表单

所有内联表单中的元素都是左对齐的。

注意:在屏幕宽度小于 576px 时为垂直堆叠,如果屏幕宽度大于等于576px时表单元素才会显示在同一个水平线上。内联表单需要在 < form > 元素上添加 .form-inline类。

```
<div class="container">
<h2>内联表单</h2>
屏幕宽度在大于等于 576px 时才会水平显示。如果小于 576px 则会生成堆
叠表单。
<form class="form-inline">
 <label for="email">Email:</label>
 <input type="email" class="form-control" id="email" placeholder="Enter
email">
 <label for="pwd">Password:</label>
 <input type="password" class="form-control" id="pwd"</pre>
placeholder="Enter password">
 <div class="form-check">
  <label class="form-check-label">
   <input class="form-check-input" type="checkbox"> Remember me
  </label>
 </div>
 <button type="submit" class="btn btn-primary">Submit</button>
</form>
</div>
```

2. 轮播

轮播创建: 以下实例创建了一个简单的图片轮播效果

```
/* Make the image fully responsive */
.carousel-inner img {
  width: 100%;
  height: 100%;
}
</style>
</head>
<body>
<div id="demo" class="carousel slide" data-ride="carousel">
<!-- 指示符 -->
data-target="#demo" data-slide-to="1">
 data-target="#demo" data-slide-to="2">
<!-- 轮播图片 -->
<div class="carousel-inner">
 <div class="carousel-item active">
src="https://atts.w3cschool.cn/attachments/knowledge/201803/24575.png
 //在每个 <div class="carousel-item"> 内添加 <div class="carousel-
caption">来设置轮播图片的描述文本
  <div class="carousel-caption">
  <h3>第一张图片描述标题</h3>
  描述文字!
 </div>
 <div class="carousel-item">
src="https://atts.w3cschool.cn/attachments/knowledge/201801/31740.png
 </div>
 <div class="carousel-item">
  <img
src="https://atts.w3cschool.cn/attachments/knowledge/201804/30601.png
 </div>
</div>
<!-- 左右切换按钮 -->
<a class="carousel-control-prev" href="#demo" data-slide="prev">
 <span class="carousel-control-prev-icon"></span>
</a>
<a class="carousel-control-next" href="#demo" data-slide="next">
 <span class="carousel-control-next-icon"></span>
</a>
</div>
</body>
</html>
```

3. 模态框

模态框(Modal)是覆盖在父窗体上的子窗体。通常,使用模态框的目的是用来显示来自一个单独的源的内容,而且可以在不离开父窗体的情况下有一些互动

```
<!DOCTYPE html>
<html>
<head>
<title>Bootstrap4 实例</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"</pre>
href="https://cdn.bootcss.com/bootstrap/4.1.0/css/bootstrap.min.css">
<script src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js">
</script>
<script
src="https://cdn.bootcss.com/popper.js/1.12.5/umd/popper.min.js">
</script>
<script
src="https://cdn.bootcss.com/bootstrap/4.1.0/js/bootstrap.min.js">
</script>
</head>
<body>
<div class="container">
<h2>模态框实例</h2>
<!-- 按钮: 用于打开模态框 -->
<button type="button" class="btn btn-primary" data-toggle="modal"</pre>
data-target="#myModal">
 打开模态框
</button>
<!-- 模态框 -->
<div class="modal fade" id="myModal">
 //我们可以通过添加.modal-sm类来创建一个小模态框,.modal-lg类可以创
建一个大模态框。
尺寸类都放在 <div>元素的 .modal-dialog 类后
 <div class="modal-dialog modal-sm">
  <div class="modal-content">
   <!-- 模态框头部 -->
   <div class="modal-header">
    <h4 class="modal-title">模态框头部</h4>
    <button type="button" class="close" data-dismiss="modal">&times;
</button>
   </div>
   <!-- 模态框主体 -->
   <div class="modal-body">
   模态框内容..
   </div>
   <!-- 模态框底部 -->
   <div class="modal-footer">
```

```
<br/>
<br/>
<br/>
<br/>
<br/>
dismiss="modal">关闭</button>
<br/>
</div>
</div>
</div>
</div>
</div>
</div>
</html>
```

4. 提示框

创建提示框

通过向元素添加 data-toggle="tooltip" 来来创建提示框。title 属性的内容为提示框显示的内容

```
<!DOCTYPE html>
<html>
<head>
 <title>Bootstrap4 实例</title>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <link rel="stylesheet"</pre>
href="https://cdn.bootcss.com/bootstrap/4.1.0/css/bootstrap.min.css">
<script src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js">
</script>
<script
src="https://cdn.bootcss.com/popper.js/1.12.5/umd/popper.min.js">
</script>
<script
src="https://cdn.bootcss.com/bootstrap/4.1.0/js/bootstrap.min.js">
</script>
</head>
<body>
<div class="container">
<h3>提示框实例</h3><br>
<a href="#" data-toggle="tooltip" title="我是提示内容!">鼠标移动到我这
</a>
<a href="#" data-toggle="tooltip" data-placement="top" title="我是提示内
容!">鼠标移动到我这</a>
<a href="#" data-toggle="tooltip" data-placement="bottom" title="我是提示
内容!">鼠标移动到我这</a>
<a href="#" data-toggle="tooltip" data-placement="left" title="我是提示内
容!">鼠标移动到我这</a>
<a href="#" data-toggle="tooltip" data-placement="right" title="我是提示内
容!">鼠标移动到我这</a>
</div>
<script>
$(document).ready(function(){
  $('[data-toggle="tooltip"]').tooltip();
});
</script>
```

```
</body>
</html>
```

5. 弹出框

创建弹出框

通过向元素添加 data-toggle="popover" 来来创建弹出框。title 属性的内容为弹出框的标题,data-content 属性显示了弹出框的文本内容

```
<!DOCTYPE html>
<html>
<head>
 <title>Bootstrap4 实例</title>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <link rel="stylesheet"</pre>
href="https://cdn.bootcss.com/bootstrap/4.1.0/css/bootstrap.min.css">
 <script src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js">
</script>
<script
src="https://cdn.bootcss.com/popper.js/1.12.5/umd/popper.min.js">
</script>
<script
src="https://cdn.bootcss.com/bootstrap/4.1.0/js/bootstrap.min.js">
</script>
</head>
<body>
<div class="container">
<h3>弹出框实例</h3>
 <a href="#" data-toggle="popover" title="弹出框标题" data-content="弹出
框内容">多次点我</a>
<a href="#" title="Header" data-toggle="popover" data-placement="top"
data-content="Content">点我</a>
<a href="#" title="Header" data-toggle="popover" data-
placement="bottom" data-content="Content">点我</a>
<a href="#" title="Header" data-toggle="popover" data-placement="left"
data-content="Content">点我</a>
<a href="#" title="Header" data-toggle="popover" data-placement="right"
data-content="Content">点我</a>
</div>
<script>
$(document).ready(function(){
 $('[data-toggle="popover"]').popover();
});
</script>
</body>
</html>
```

关闭弹出框

默认情况下,弹出框在再次点击指定元素后就会关闭,你可以使用 data-trigger="focus" 属性来设置在鼠标点击元素外部区域来关闭弹出框

5. 滚动监听

滚动监听创建

然后添加 data-target 属性,它的值为导航栏的 id 或 class (.navbar)。这样就可以联系上可滚动区域。 向您想要监听的元素(通常是 body)添加 data-spy="scroll"。 注意可滚动项元素上的 id(< div id="section1" >)必须匹配导航栏上的链接选项(< a href="#section1" >)。 可选项data-offset 属性用于计算滚动位置时,距离顶部的偏移像素。 默认为 10 px

```
<!DOCTYPE html>
<html>
<head>
<title>Bootstrap4 实例</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"</pre>
href="https://cdn.bootcss.com/bootstrap/4.1.0/css/bootstrap.min.css">
<script src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js">
</script>
<script
src="https://cdn.bootcss.com/popper.js/1.12.5/umd/popper.min.js">
</script>
<script
src="https://cdn.bootcss.com/bootstrap/4.1.0/js/bootstrap.min.js">
</script>
<style>
body {
  position: relative;
</style>
</head>
<body data-spy="scroll" data-target=".navbar" data-offset="50">
<nav class="navbar navbar-expand-sm bg-dark navbar-dark fixed-top">
 ul class="navbar-nav">
 class="nav-item">
  <a class="nav-link" href="#section1">Section 1</a>
 cli class="nav-item">
  <a class="nav-link" href="#section2">Section 2</a>
 cli class="nav-item">
  <a class="nav-link" href="#section3">Section 3</a>
 class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" id="navbardrop" data-
toggle="dropdown">
   Section 4
  </a>
  <div class="dropdown-menu">
   <a class="dropdown-item" href="#section41">Link 1</a>
   <a class="dropdown-item" href="#section42">Link 2</a>
```

```
</div>
 </nav>
<div id="section1" class="container-fluid bg-success" style="padding-
top:70px;padding-bottom:70px">
<h1>Section 1</h1>
学编程,从W3Cschool开始! 
</div>
<div id="section2" class="container-fluid bg-warning" style="padding-
top:70px;padding-bottom:70px">
<h1>Section 2</h1>
学编程,从W3Cschool开始! 
</div>
<div id="section3" class="container-fluid bg-secondary" style="padding-
top:70px;padding-bottom:70px">
<h1>Section 3</h1>
学编程,从W3Cschool开始! 
</div>
<div id="section41" class="container-fluid bg-danger" style="padding-
top:70px;padding-bottom:70px">
<h1>Section 4 Submenu 1</h1>
学编程,从W3Cschool开始! 
</div>
<div id="section42" class="container-fluid bg-info" style="padding-
top:70px;padding-bottom:70px">
<h1>Section 4 Submenu 2</h1>
>学编程,从W3Cschool开始! 
</div>
</body>
</html>
```

6. 多媒体对象

基础多媒体对象创建:要创建一个多媒体对象,可以在容器元素上添加.media类,然后将多媒体内容放到子容器上,子容器需要添加.media-body类,然后添加外边距,内边距等效果

```
<!DOCTYPE html>
<html>
<head>
<title>Bootstrap4 实例</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
link rel="stylesheet"
href="https://cdn.bootcss.com/bootstrap/4.1.0/css/bootstrap.min.css">
<script src="https://cdn.bootcss.com/jquery/3.2.1/jquery.min.js">
</script>
<script
src="https://cdn.bootcss.com/popper.js/1.12.5/umd/popper.min.js">
</script>
</script>
```

```
<script
src="https://cdn.bootcss.com/bootstrap/4.1.0/js/bootstrap.min.js">
</script>
</head>
<body>
<div class="container mt-3">
<h2>多媒体对象</h2>
使用 .media 和 .media-body 类创建多媒体对象:
<div class="media border p-3">
 <img
src="https://www.w3cschool.cn/attachments/cover/cover_bootstrap4.jpg"
alt="John Doe" class="mr-3 mt-3 rounded-circle" style="width:60px;">
 <div class="media-body">
  <h4>bootstrap4教程</h4>
  >Bootstrap4 目前是 Bootstrap 的最新版本,是一套用于 HTML、CSS 和
JS 开发的开源工具集。
 </div>
</div>
</div>
</body>
</html>
```